Multigrid for nonlinear problems: an overview

Van Emden Henson

Center for Applied Scientific Computing
Lawrence Livermore National Laboratory

vhenson@llnl.gov

http://www.casc.gov/CASC/people/henson

January 23, 2003

This work was performed, in part, under the auspices of the United States
Department of Energy by University of California Lawrence Livermore
National Laboratory under contract number W-7405-Eng-48.

Outline

- Multigrid: a 30-second introduction
- The scalar Newton's method
- Newton's method for systems
- Multigrid for Newton's method: Newton-MG
- Nonlinear multigrid: full approximation storage (FAS)
- Numerical examples of Newton-MG and FAS

The 1-d Model Problem

- Poisson's equation: $-\Delta u = f$ in [0,1], with boundary conditions $u(0) = u(1) = 0$.
- Discretized as:

$$-u_{i-1} + 2u_i - u_{i+1} = h^2 f_i \qquad u_0 = u_N = 0$$

- Leads to the Matrix equation $Au = f$, where

$$A = \begin{pmatrix} 2 & -1 & & & & \\ -1 & 2 & -1 & & & \\ & -1 & 2 & -1 & & \\ & & & & & \\ & & & -1 & 2 & -1 \\ & & & & -1 & 2 \end{pmatrix}, \quad u = \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ \\ u_{N-2} \\ u_{N-1} \end{pmatrix}, \quad f = \begin{pmatrix} f_1 \\ f_2 \\ f_3 \\ \\ f_{N-2} \\ f_{N-1} \end{pmatrix}$$

Weighted Jacobi Relaxation

- Consider the iteration:

$$u_i^{(new)} \leftarrow (1-\omega)\, u_i^{(old)} + \frac{\omega}{2}\left(u_{i-1}^{(old)} + u_{i+1}^{(old)} + h^2 f_i \right)$$

- Letting ***A = D-L-U,*** the matrix form is:

$$u^{(new)} = \left[(1-\omega) I + \omega D^{-1}(L+U) \right] u^{(old)} + \omega h^2 D^{-1} f$$

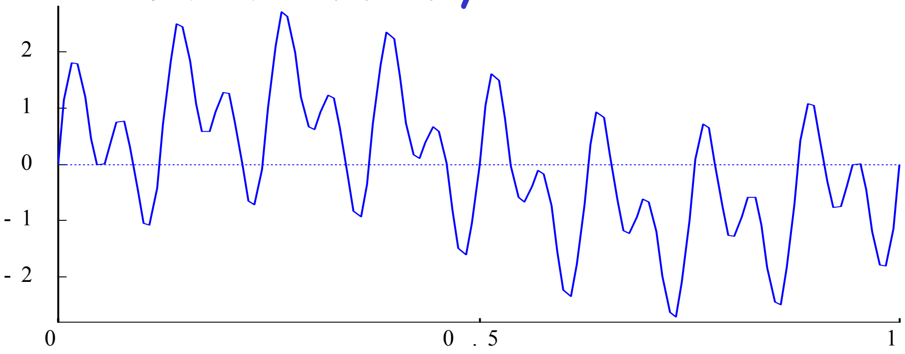
$$= R_\omega u^{(old)} + \omega h^2 D^{-1} f \qquad .$$

- It is easy to see that if $e \equiv u^{(exact)} - u^{(approx)}$, then

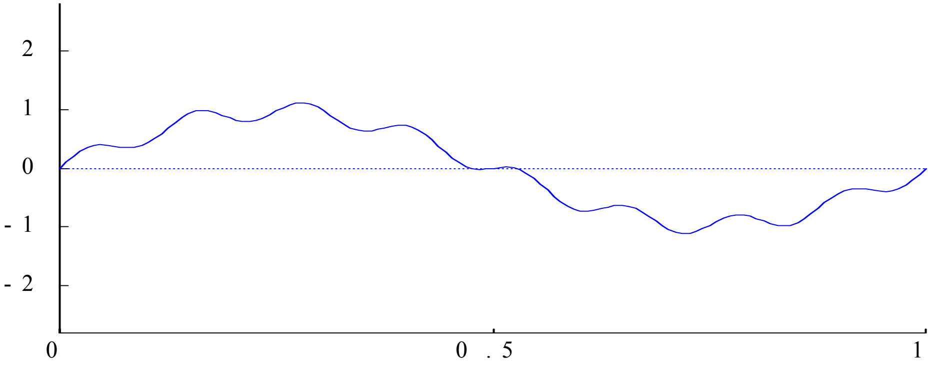
$$e^{(new)} = R_\omega e^{(old)}$$

Relaxation Smoothes the Error

- Initial error,



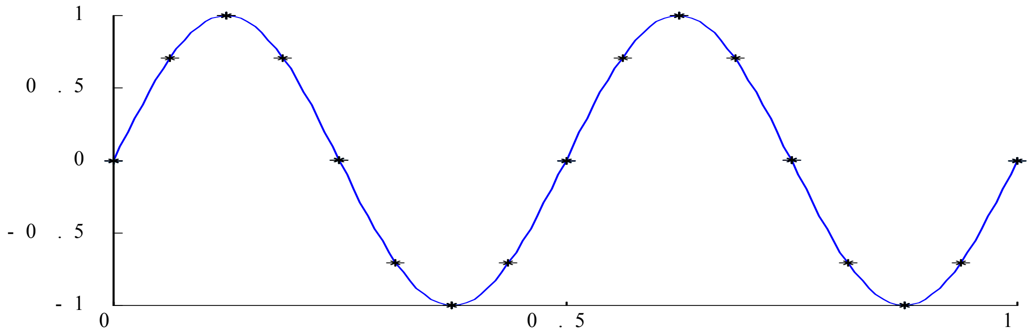
- Error after 35 iteration sweeps:



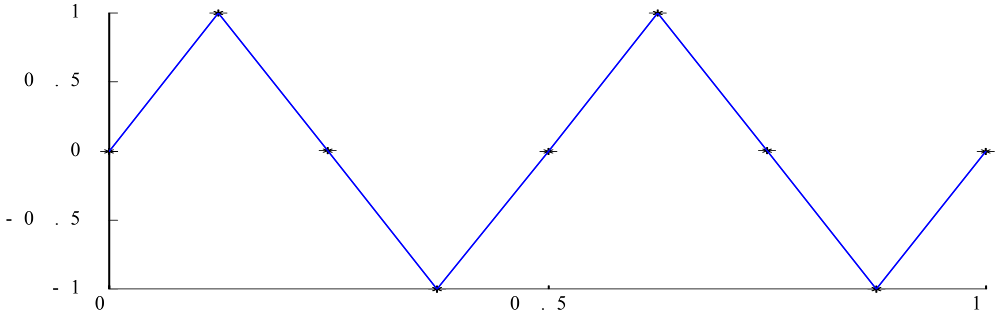
Many relaxation schemes have the smoothing property, where oscillatory modes of the error are eliminated effectively, but smooth modes are damped very slowly.

Smooth error can be represented accurately on a coarse grid

- A smooth function:



- Can be represented by linear interpolation from a coarser grid:



On the coarse grid, the smooth error appears to be relatively higher in frequency: in the example it is the 4-mode, out of a possible 16, on the fine grid, 1/4 the way up the spectrum. On the coarse grid, it is the 4-mode out of a possible 8, hence it is 1/2 the way up the spectrum.

Relaxation will be more effective on this mode if done on the coarser grid!!

Coarse-grid Correction

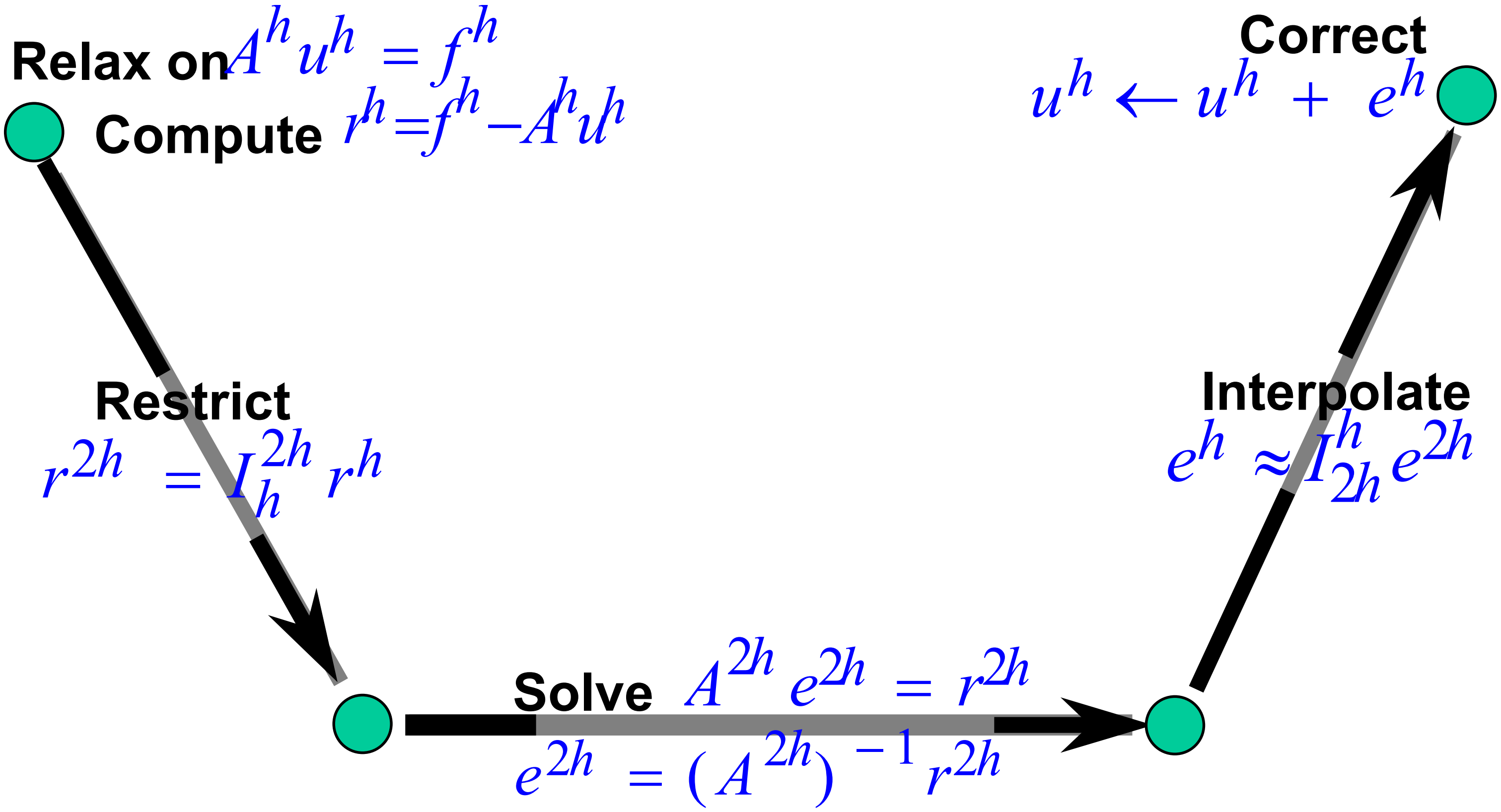
- Perform relaxation on $A^h u^h = f^h$ on fine grid until error is smooth.

- Compute residual, $r^h = f^h - A^h u^h$ and transfer to the coarse grid $r^{2h} = I_h^{2h} r^h$.

- Solve the coarse-grid residual equation to obtain the error:
  $$A^{2h} e^{2h} = r^{2h}, \quad \therefore \; e^{2h} = (A^{2h})^{-1} r^{2h}$$

- Interpolate the error to the fine grid and correct the fine-grid solution:
  $$u^h \leftarrow u^h + I_{2h}^h e^{2h}$$

Coarse-grid Correction



Relax on $A^h u^h = f^h$

Compute $r^h = f^h - A^h u^h$

Restrict $r^{2h} = I_h^{2h} r^h$

Solve $A^{2h} e^{2h} = r^{2h}$

$e^{2h} = (A^{2h})^{-1} r^{2h}$

Interpolate $e^h \approx I_{2h}^h e^{2h}$

Correct $u^h \leftarrow u^h + e^h$

Tools Needed

- Interpolation and restriction operators:

$$I_{2h}^{h} = \begin{pmatrix} 0.5 & & \\ 1.0 & & \\ 0.5 & 0.5 & \\ & 1.0 & \\ & 0.5 & 0.5 \\ & & 1.0 \\ & & 0.5 \end{pmatrix}, \qquad I_{h}^{2h} = \begin{pmatrix} 0 & 1 & 0 & & & & \\ & & 0 & 1 & 0 & & \\ & & & & 0 & 1 & 0 \end{pmatrix}, \qquad I_{h}^{2h} = \begin{pmatrix} 0.25 & 1.0 & 0.25 & & & & \\ & & 0.25 & 1.0 & 0.25 & & \\ & & & & 0.25 & 1.0 & 0.25 \end{pmatrix}$$

Linear Interpolation

Injection

Full-weighting

- Coarse-grid Operator A^{2h} . Two methods:

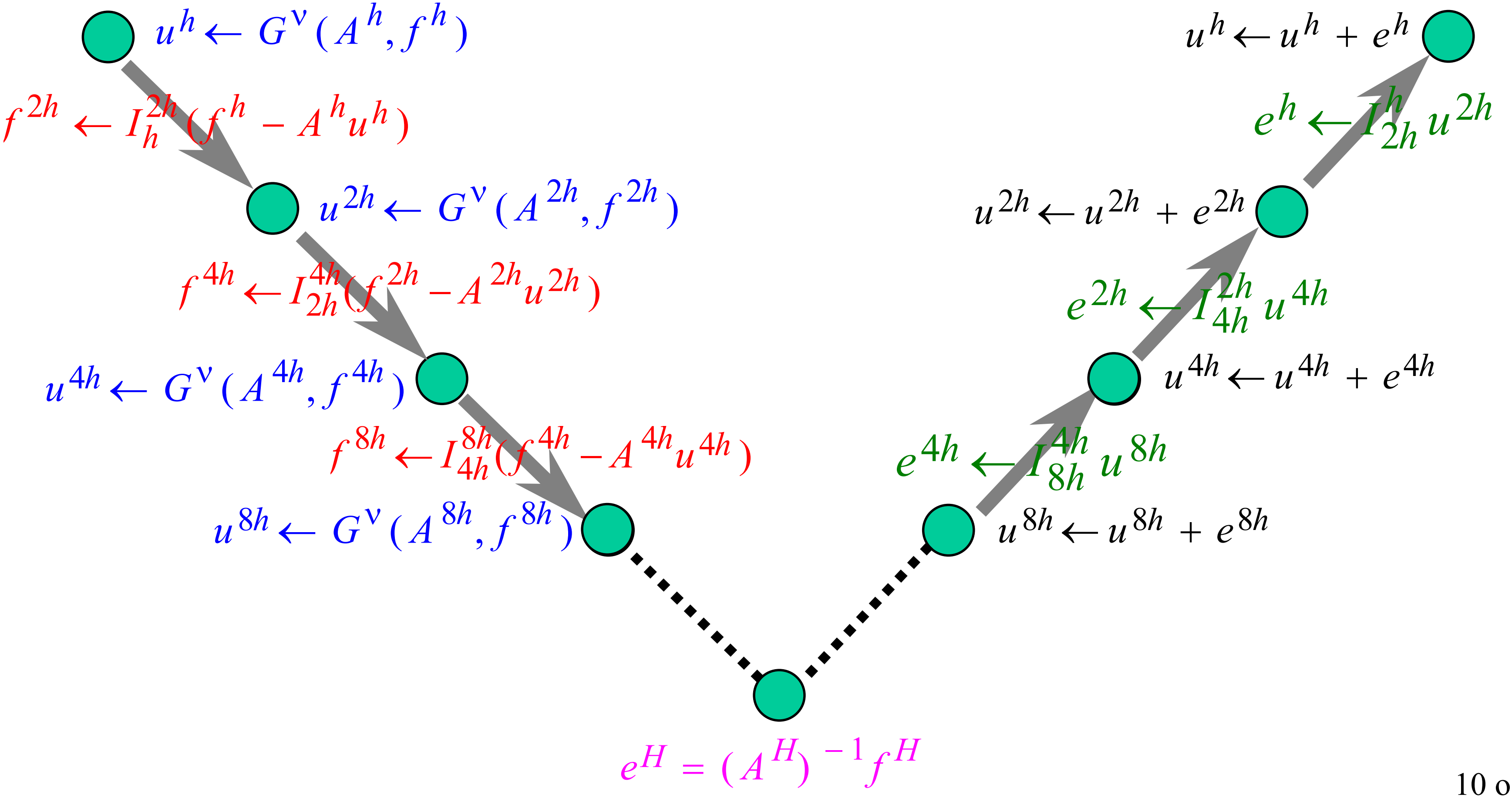
  (1) Discretize equation at larger spacing

  (2) Use Galerkin Formula:

$$A^{2h} = I_{h}^{2h} A^{h} I_{2h}^{h}$$

Recursion: the (ν ,0) V-cycle

- Major question: How do we “solve” the coarse-grid residual equation? Answer: *recursion*!



$u^h \leftarrow G^\nu(A^h, f^h)$

$f^{2h} \leftarrow I_h^{2h}(f^h - A^h u^h)$

$u^{2h} \leftarrow G^\nu(A^{2h}, f^{2h})$

$f^{4h} \leftarrow I_{2h}^{4h}(f^{2h} - A^{2h} u^{2h})$

$u^{4h} \leftarrow G^\nu(A^{4h}, f^{4h})$

$f^{8h} \leftarrow I_{4h}^{8h}(f^{4h} - A^{4h} u^{4h})$

$u^{8h} \leftarrow G^\nu(A^{8h}, f^{8h})$

$e^H = (A^H)^{-1} f^H$

$u^{8h} \leftarrow u^{8h} + e^{8h}$

$e^{4h} \leftarrow I_{8h}^{4h} u^{8h}$

$u^{4h} \leftarrow u^{4h} + e^{4h}$

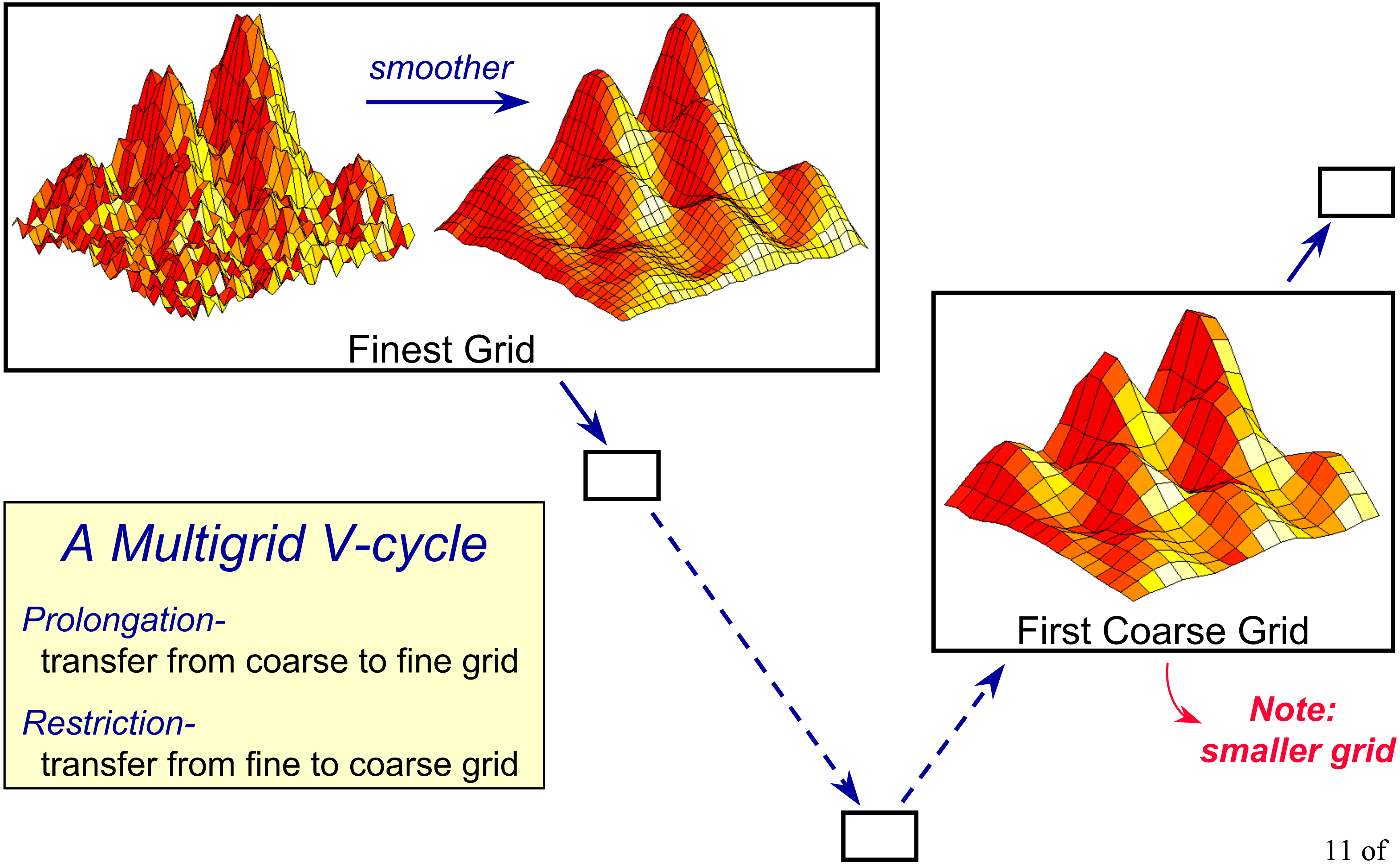
$e^{2h} \leftarrow I_{4h}^{2h} u^{4h}$

$u^{2h} \leftarrow u^{2h} + e^{2h}$

$e^h \leftarrow I_{2h}^h u^{2h}$

$u^h \leftarrow u^h + e^h$

Multigrid uses coarse grids to damp out smooth error components



A Multigrid V-cycle

Prolongation-
transfer from coarse to fine grid

Restriction-
transfer from fine to coarse grid

Nested Iteration

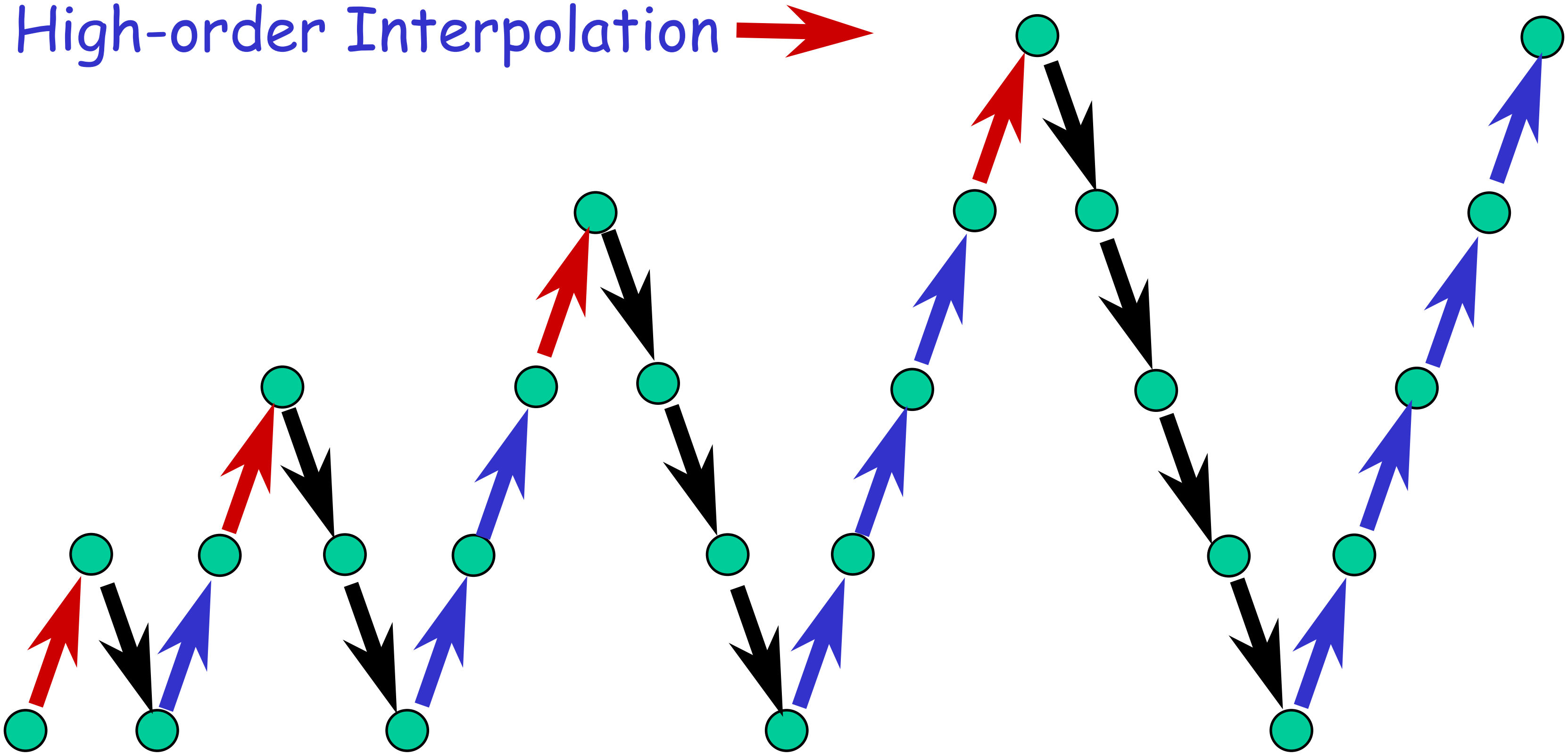
- *Idea:* It's cheaper to solve a problem (i.e., takes fewer iterations) if the initial guess is good.

- How to get a good initial guess:
  - Interpolate coarse solution to fine grid.
  - "Solve" the problem on the coarse grid first.
  - Use interpolated coarse solution as initial guess on fine grid.

- **Combined with the V-cycle as the solver this defines the Full Multigrid (FMG) cycle.**

Full Multigrid (FMG)

- Restriction →
- Interpolation →
- High-order Interpolation →



Nonlinear Problems

- How should we approach the nonlinear system

$$A(u) = f$$

and can we use multigrid to solve such a system?

- A fundamental relation that multigrid relies on, the residual equation

$$Au - Av = f - Av \Rightarrow Ae = r$$

does not hold, since, if $A(u)$ is a nonlinear operator,

$$A(u) - A(v) \neq A(e)$$

The Nonlinear Residual Equation

- We still base our development around the residual equation, now the nonlinear residual equation:

$$A(u) = f$$

$$A(u) - A(v) = f - A(v)$$

$$\boxed{A(u) - A(v) = r}$$

- How can we use this equation as the basis for a solution method?

Let's consider Newton's Method

- The best known and most important method for solving nonlinear equations!
- We wish to solve $F(x) = 0$.
- Expand F in a Taylor series about x:

$$F(x+s) = F(x) + sF'(x) + s^2 F''(\xi)$$

- Dropping higher order terms, if $x+s$ is a solution,

$$0 = F(s) + sF'(x) \qquad \therefore \qquad s = -F(x) / F'(x)$$

- Hence, we develop an iteration

$$x \leftarrow x - \frac{F(x)}{F'(x)}$$

Newton's method for systems

- We wish to solve the system A(u) = 0. In vector form this is

$$A(u) = \begin{pmatrix} f_1(u_1, u_2, \ldots, u_N) \\ f_2(u_1, u_2, \ldots, u_N) \\ \ddots \\ f_N(u_1, u_2, \ldots, u_N) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \ddots \\ 0 \end{pmatrix}$$

- Expanding $A(v+e)$ in a Taylor series about v :

$$A(v+e) = A(v) + J(v)\,e + \text{higher order terms}$$

Newton for systems (cont.)

- Where *J(v)* is the Jacobian system

$$J(v) = \left.\begin{pmatrix} \frac{\partial f_1}{\partial u_1} & \frac{\partial f_1}{\partial u_2} & \cdots & \frac{\partial f_1}{\partial u_N} \\ \frac{\partial f_2}{\partial u_1} & \frac{\partial f_2}{\partial u_2} & \cdots & \frac{\partial f_2}{\partial u_N} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_N}{\partial u_1} & \frac{\partial f_N}{\partial u_2} & \cdots & \frac{\partial f_N}{\partial u_N} \end{pmatrix}\right|_{u=v}$$

- If *u=v+e* is a solution, *0 = A(v) + J(v) e* and

$$e = -[J(v)]^{-1} A(v)$$

- Leading to the iteration

$$\boxed{v \leftarrow v - [J(v)]^{-1} A(v)}$$

Newton's method in terms of the residual equation

- The nonlinear residual equation is

$$A(v+e) - A(v) = r$$

- Expanding $A(v+e)$ in a two-term Taylor series about v:

$$A(v) + J(v)\, e - A(v) = r$$

$$J(v)\, e = r$$

- Newton's method is thus:

$$r = f - A(v)$$

$$v \leftarrow v + [J(v)]^{-1} r$$

How does multigrid fit in?

- One obvious method is to use multigrid to solve $J(v)e = r$ at each iteration step. This method is called Newton-multigrid and can be very effective.

- However, we would like to use multigrid ideas to treat the nonlinearity directly.

- Hence, we need to specialize the multigrid components (relaxation, grid transfers, coarsening) for the nonlinear case.

What is nonlinear relaxation?

- Several of the common relaxation schemes have nonlinear counterparts. For A(u)=f, we describe the nonlinear Gauss-Seidel iteration:
  - For each j=1, 2, …, N
    - Set the *jth* component of the residual to zero and solve for v_j. That is, solve $(A(v))_j = f_j$.
- Equivalently,
  - For each j=1, 2, …, N
    - Find $s \in \Re$ such that

$$(A(v + s\,\varepsilon_j))_j = f_j$$

where ε_j is the canonical *jth* unit basis vector

How is nonlinear Gauss-Seidel done?

- Each $(A(v))_j = f_j$ is a nonlinear scalar equation for v_j. We use the scalar Newton's method to solve!

- Example: $-u''(x) + u(x)\, e^{u(x)} = f$, may be discretized so that $(A(v))_j = f_j$ is given by

$$\frac{-v_{j-1} + 2v_j - v_{j+1}}{h^2} + v_j e^{v_j} = f_j \qquad 1 \le j \le N-1$$

- Newton iteration for v_j is given by

$$v_j \leftarrow v_j - \frac{\dfrac{-v_{j-1} + 2v_j - v_{j+1}}{h^2} + v_j e^{v_j} - f_j}{\dfrac{2}{h^2} + e^{v_j}(1 + v_j)}$$

How do we do coarsening for nonlinear multigrid?

- Recall the nonlinear residual equation

$$A(v+e) - A(v) = r$$

- In multigrid, we obtain an approximate solution v^h on the fine grid, then solve the residual equation on the coarse grid.

- The nonlinear residual equation on Ω^{2h} appears as

$$A^{2h}(v^{2h} + e^{2h}) - A^{2h}(v^{2h}) = r^{2h}$$

Look at the coarse residual equation

- We must evaluate the quantities on Ω^{2h} in

$$A^{2h}(v^{2h}+e^{2h})-A^{2h}(v^{2h})=r^{2h}$$

- Given v^{h}, a fine-grid approximation, we restrict the residual to the coarse grid

$$r^{2h}=I_h^{2h}(f^{h}-A^{h}(v^{h}))$$

- For v^{2h} we restrict v^{h} by $v^{2h}=I_h^{2h}v^{h}$
- Thus,

$$\boxed{A^{2h}(I_h^{2h}v^{h}+e^{2h})=A^{2h}(I_h^{2h}v^{h})+I_h^{2h}(f^{h}-A^{h}(v^{h}))}$$

We've obtained a coarse-grid equation of the form $A^{2h}(u^{2h}) = f^{2h}$.

- Consider the coarse-grid residual equation:

$$A^{2h}(\underbrace{I_h^{2h}v^h + e^{2h}}_{u^{2h}}) = \underbrace{A^{2h}(I_h^{2h}v^h) + I_h^{2h}(f^h - A^h(v^h))}_{f^{2h}}$$

coarse-grid unknown

All quantities are known

- We solve $A^{2h}(u^{2h}) = f^{2h}$ for $u^{2h} = I_h^{2h}v^h + e^{2h}$ and obtain

$$\boxed{e^{2h} = u^{2h} - I_h^{2h}v^h}$$

- We then apply the correction:

$$v^h = v^h + I_{2h}^h e^{2h}$$

FAS, the Full Approximation Scheme, two grid form

- Perform nonlinear relaxation on $A^h(u^h) = f^h$ to obtain an approximation v^h .

- Restract the approximation and its residual

$$v^{2h} = I_h^{2h} v^h \qquad r^{2h} = I_h^{2h} (f^h - A(v^h))$$

- Solve the coarse-grid residual problem

$$A^{2h}(u^{2h}) = A^{2h}(v^{2h}) + r^{2h}$$

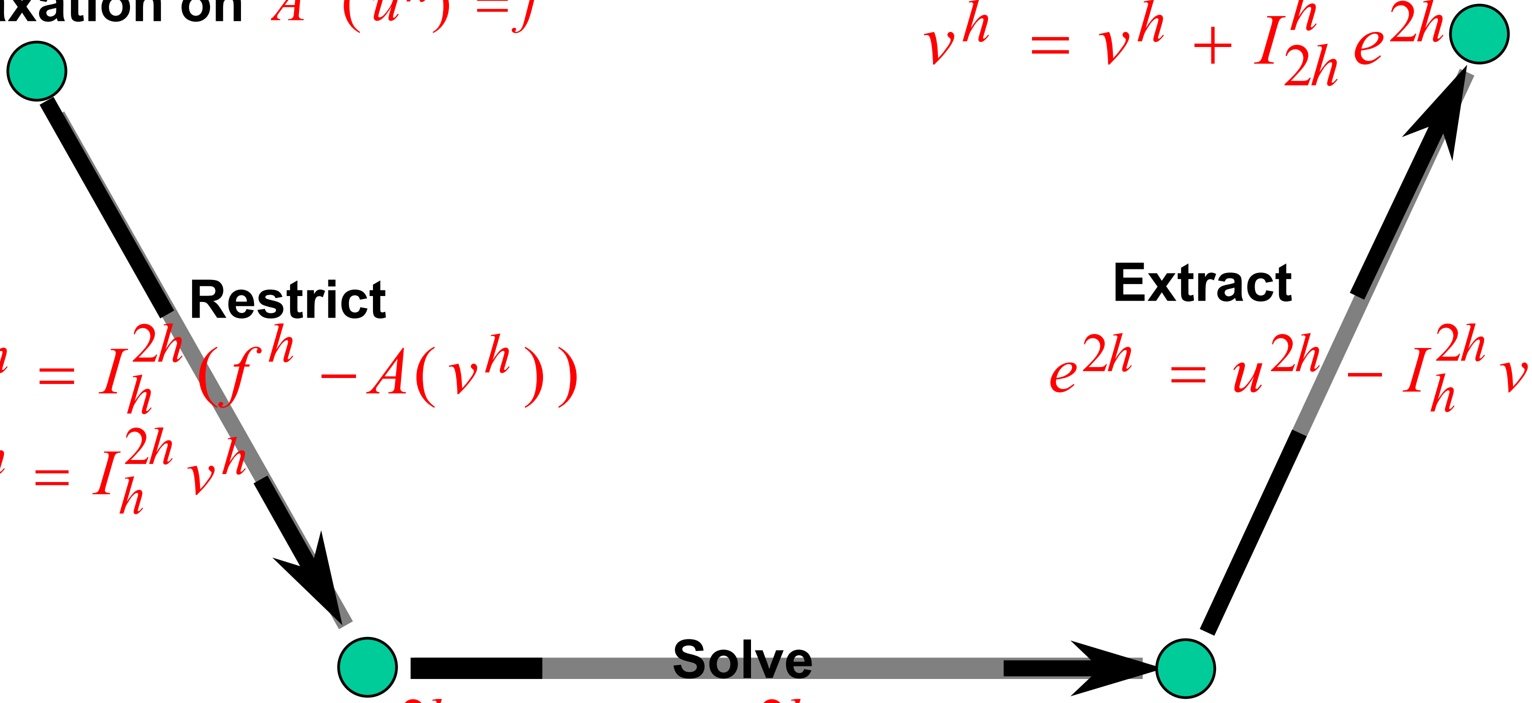
- Extract the coarse-grid error

$$e^{2h} = u^{2h} - v^{2h}$$

- Interpolate and apply the correction

$$v^h = v^h + I_{2h}^h e^{2h}$$

FAS, the Full Approximation Scheme, two grid form



Nonlinear Relaxation on $A^h(u^h) = f^h$

Restrict

$$r^{2h} = I_h^{2h}(f^h - A(v^h))$$

$$v^{2h} = I_h^{2h} v^h$$

Solve

$$A^{2h}(u^{2h}) = A^{2h}(v^{2h}) + r^{2h}$$

Extract

$$e^{2h} = u^{2h} - I_h^{2h} v^h$$

Interpolate & Correct

$$v^h = v^h + I_{2h}^h e^{2h}$$

A few observations about FAS

- If *A* is a linear operator then FAS reduces directly to the linear two-grid correction scheme.

- A fixed point of FAS is an exact solution to the fine-grid problem and an exact solution to the fine-grid problem is a fixed point of the FAS iteration.

A few observations about FAS, continued

- The FAS coarse-grid equation can be written as

$$A^{2h}(u^{2h}) = f^{2h} + \tau_h^{2h}$$

where τ_h^{2h} is the so-called tau correction.

- In general, since $\tau_h^{2h} \neq 0$, the solution u^{2h} to the FAS coarse-grid equation is not the same as the solution to the original coarse-grid problem .

$$A^{2h}(u^{2h}) = f^{2h}$$

- The tau correction may be viewed as a way to alter the coarse-grid equations to enhance their approximation properties.

Still more observations about FAS

- FAS may be viewed as an inner and outer iteration: the outer iteration is the coarse-grid correction, the inner iteration the relaxation method.

- A true multilevel FAS process is recursive, using FAS to solve the nonlinear Ω^{2h} problem using Ω^{4h}. Hence, FAS is generally employed in a V- or W-cycling scheme.

And yet more observations about FAS!

- For linear problems we use FMG to obtain a good initial guess on the fine grid. Convergence of nonlinear iterations depends critically on having a good initial guess.

- When FMG is used for nonlinear problems the interpolant $I_{2h}^{h} u^{2h}$ is generally accurate enough to be in the basin of attraction of the fine-grid solver.

- Thus, one FMG cycle, whether FAS, Newton, or Newton-multigrid is used on each level, should provide a solution accurate to the level of discretization, unless the nonlinearity is extremely strong.

Intergrid transfers for FAS

- Generally speaking, the standard operators (linear interpolation, full weighting) work effectively in FAS schemes.

- In the case of strongly nonlinear problems, the use of higher-order interpolation (e.g., cubic interpolation) may be beneficial.

- For an FMG scheme, where $I_{2h}^{h} u^{2h}$ is the interpolation of a coarse-grid solution to become a fine-grid initial guess, higher-order interpolation is always advised.

What is $A^{2h}(u^{2h})$ in FAS?

- As in the linear case, there are two basic possibilities:

- $A^{2h}(u^{2h})$ is determined by discretizing the nonlinear operator *A(u)* in the same fashion as was employed to obtain $A^{h}(u^{h})$, except that the coarser mesh spacing is used.

- $A^{2h}(u^{2h})$ is determined from the Galerkin condition

$$A^{2h}(u^{2h}) = I_h^{2h} A^{h}(u^{h}) I_{2h}^{h}$$

where the action of the Galerkin product can be captured in an implementable formula.

- The first method is usually easier, and more common.

Nonlinear problems: an example

- Consider

$$-\Delta u(x,y) + \gamma u(x,y)\, e^{u(x,y)} = f(x,y)$$

  on the unit square [0,1] × [0,1] with homogeneous Dirichlet boundary conditions and a regular Cartesian grid.

- Suppose the exact solution is

$$u(x,y) = (x^2 - x^3)\sin(3\pi y)$$

Discretization of nonlinear example

- The operator can be written (sloppily) as

$$\underbrace{\frac{1}{h^2}\begin{pmatrix} & -1 & \\ -1 & 4 & -1 \\ & -1 & \end{pmatrix} u^h_{i,j} + \gamma u^h_{i,j} e^{u^h_{i,j}}}_{A^h(u^h)} = f_{i,j}$$

- The relaxation is given by

$$u^h_{i,j} \leftarrow u^h_{i,j} - \frac{(A^h(u^h))_{i,j} - f_{i,j}}{\frac{4}{h^2} + \gamma(1 + u^h_{i,j})\, e^{u_{i,j}}}$$

FAS and Newton's method on

$$-\Delta u(x,y) + \gamma u(x,y)\, e^{u(x,y)} = f(x,y)$$

- FAS

	γ			
	1	10	100	1000
convergence factor	0.135	0.124	0.098	0.072
number of FAS cycles	12	11	11	10

- Newton's Method

	γ			
	1	10	100	1000
convergence factor	4.00E-05	7.00E-05	3.00E-04	2.00E-04
number of Newton iterations	3	3	3	4

Newton, Newton-MG, and FAS on

$$-\Delta u(x,y) + \gamma u(x,y)\, e^{u(x,y)} = f(x,y)$$

- Newton uses exact solve, Newton-MG is inexact Newton with a fixed number of inner V(2,1)-cycles the Jacobian problem, overall stopping criterion $\|r\|_2 < 10^{-10}$

	Outer	Inner	
Method	iterations	iterations	Megaflops
Newton	3		1660.6
Newton-MG	3	20	56.4
Newton-MG	4	10	38.5
Newton-MG	5	5	25.1
Newton-MG	10	2	22.3
Newton-MG	19	1	24.6
FAS	11		27.1

Comparing FMG-FAS and FMG-Newton

$$-\Delta u(x,y) + \gamma u(x,y)\, e^{u(x,y)} = f(x,y)$$

We will do one FMG cycle using a single FAS V(2,1) -cycle as the “solver” at each new level. We then follow that with sufficiently many FAS V(2,1)-cycles as is necessary to obtain $||r|| < 10^{-10}$.

Next, we will do one FMG cycle using a Newton-multigrid step at each new level (with a single linear V(2,1)-cycle as the Jacobian “solver.”) We then follow that with sufficiently many Newton-multigrid steps as is necessary to obtain $||r|| < 10^{-10}$.

Comparing FMG-FAS and FMG-Newton

$$-\Delta u(x,y) + \gamma u(x,y)\, e^{u(x,y)} = f(x,y)$$

Cycle	$\|r^h\|$	$\|e^h\|$	Mflops	$\|r^h\|$	$\|e^h\|$	Mflops	Cycle
FMG-FAS	1.10E-02	2.00E-05	3.1	1.06E-02	2.50E-05	2.4	FMG-Newton
FAS V	6.80E-04	2.40E-05	5.4	6.70E-04	2.49E-05	4.1	Newton-MG
FAS V	5.00E-05	2.49E-05	7.6	5.10E-05	2.49E-05	5.8	Newton-MG
FAS V	3.90E-06	2.49E-05	9.9	6.30E-06	2.49E-05	7.5	Newton-MG
FAS V	3.20E-07	2.49E-05	12.2	1.70E-06	2.49E-05	9.2	Newton-MG
FAS V	3.00E-08	2.49E-05	14.4	5.30E-07	2.49E-05	10.9	Newton-MG
FAS V	2.90E-09	2.49E-05	16.7	1.70E-07	2.49E-05	12.6	Newton-MG
FAS V	3.00E-10	2.49E-05	18.9	5.40E-08	2.49E-05	14.3	Newton-MG
FAS V	3.20E-11	2.49E-05	21.2	1.70E-08	2.49E-05	16.0	Newton-MG
				5.50E-09	2.49E-05	17.7	Newton-MG
				1.80E-09	2.49E-05	19.4	Newton-MG
				5.60E-10	2.49E-05	21.1	Newton-MG
				1.80E-10	2.49E-05	22.8	Newton-MG
				5.70E-11	2.49E-05	24.5	Newton-MG

Conclusions

- Multigrid can be used for nonlinear problems either by using an inexact Newton's method with multigrid as the Jacobian system solver or by using FAS to perform multigrid processing directly on the nonlinear system.

- The choice of Newton-MG versus FAS is best made with consideration of the cost of nonlinear evaluations; performance of the two methods is often similar.

- In either case FMG type cycling should be used.